

**Verification of Translation**

I, Robin Holding, having an office at 948 15th Street, #4, Santa Monica, CA 90403-3134, hereby state that I am well acquainted with both the English and French languages and that to the best of my knowledge and ability, the appended document is a true and faithful translation of

**Int'l. Patent Application No.** PCT/FR00/01047

**Filed on** April 20, 2000

**In the name of** Louis GOUBIN and Jacques PATARIN

I further declare that the above statement is true; and further, that this statement is made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent resulting therefrom.

December 14, 2000

Date

Robin Holding  
Robin Holding

3/PRT3

09/720085  
0001 Rec'd PCT/PTO 20 DEC 2000

1

Procédé de vérification de signature  
ou d'authentification

La présente invention concerne un procédé  
5 permettant de rendre plus efficace, en temps de calcul, en  
RAM et ROM nécessaires, la vérification d'une signature ou  
d'une authentification asymétrique requérant quelques  
multiplications modulo  $n$  ou des grands nombres.  
Les algorithmes de signature ou d'authentification RSA et  
10 Rabin sont des exemples permettant la mise en œuvre de ce  
procédé.

Le procédé est plus particulièrement adapté en vue  
d'une mise en œuvre dans le cas d'un ordinateur, par  
exemple un ordinateur personnel désigné par PC, qui génère  
15 une signature ou une authentification au moyen d'une clé  
secrète qui doit ensuite être vérifiée par une carte à  
microcalculateur. Le microcalculateur effectue cette  
vérification au moyen d'une clé publique. Il dispose de  
relativement peu de puissance en comparaison du PC.

20 Par "carte à microcalculateur", on entend un  
microcontrôleur monolithique standard, avec mémoire  
incorporée.

Actuellement la majorité des algorithmes à clé  
publique utilisés dans le monde effectuent des calculs  
25 modulo de "grands nombres". Par "grands nombres", on  
désigne des nombres entiers positifs et d'au moins 320  
bits. Pour des raisons de sécurité, la communauté  
scientifique recommande même actuellement d'utiliser des  
nombres d'au moins 512 bits, voire 1024 bits pour la  
30 plupart des algorithmes, par exemple pour les algorithmes  
RSA ou Rabin.

Actuellement les cartes à microcalculateur sont  
amenées à dialoguer avec des ordinateurs ayant des

capacités de calcul bien plus importantes qu'elles-mêmes. De plus, pour des raisons de coût, on utilise souvent des cartes à microcalculateur sans coprocesseur arithmétique, et avec des ressources en mémoire (ROM, RAM et EEPROM) très limitées. De ce fait, les calculs normalement requis pour réaliser une vérification d'authentification, ou une vérification de signature à clé publique, utilisant des calculs modulo de grands nombres sont souvent très longs, voire impossible faute de mémoire suffisante, si l'on utilise les descriptions traditionnelles des algorithmes cryptographiques.

Dans la suite de la description on désigne par :

- "prouveur" : l'entité qui veut être authentifiée, ou qui produit une signature. Elle effectue pour cela des calculs faisant intervenir la clé secrète de l'algorithme asymétrique utilisé. Il s'agira par exemple d'un ordinateur de type PC.
- "vérifieur" : l'entité qui vérifie l'authentification, ou qui vérifie la validité d'une signature. Elle effectue pour cela des calculs faisant intervenir uniquement la clé publique de l'algorithme cryptographique asymétrique utilisé. Il s'agira par exemple d'une carte à microcalculateur.

La présente invention a pour objet la mise en œuvre d'un procédé de vérification de signature et d'authentification permettant de remédier aux inconvénients précités inhérents à la capacité de calcul plus limitée d'une entité vérifieur, constituée par une carte à microcalculateur, vis-à-vis d'une entité prouveur, tel qu'un ordinateur personnel ou autre muni d'un dispositif lecteur de carte.

Un autre objet de la présente invention est en conséquence une simplification des opérations de calcul de

certaines réductions modulaires du vérifieur grâce à la mise en œuvre de calculs supplémentaires du prouveur, la tâche du vérifieur étant ainsi simplifiée en l'absence de tout affaiblissement de la sécurité théorique de l'ensemble.

Le procédé de vérification de signature respectivement d'authentification au moyen d'un processus de calcul cryptographique asymétrique à clé privée et à clé publique, objet de la présente invention, ce procédé étant conduit entre une entité "prouveur" et une entité "vérifieur", l'entité prouveur effectuant des calculs cryptographiques à partir de la clé privée en vue d'effectuer un calcul de signature, respectivement une valeur d'authentification, et l'entité vérifieur à partir de cette valeur transmise effectuant des calculs cryptographiques à partir de cette clé publique en vue de procéder à cette vérification de signature, respectivement à cette authentification, les opérations de calcul cryptographique mettant en œuvre le calcul de multiplications modulo  $n$  ou des grands nombres, est remarquable en ce que, pour un processus de calcul cryptographique mettant en œuvre une clé publique, constituée par un exposant public  $e$  et un modulo public  $n$ , et une clé privée constituée par un exposant privé,  $d$ , ce procédé consiste à calculer, au niveau de l'entité prouveur, au moins une valeur de prévalidation et à transmettre de l'entité prouveur à l'entité vérifieur cette au moins une valeur de prévalidation, permettant à l'entité vérifieur d'effectuer au moins une réduction modulaire en l'absence de toute opération de division pour cette réduction modulaire.

Le procédé, objet de la présente invention, s'applique dans le cadre de tout dialogue ou protocole

d'échange de messages entre une entité prouveur telle qu'un ordinateur personnel et une entité vérifieur telle qu'une carte à microcalculateur, en particulier dans le cadre de transactions bancaires, de contrôle d'accès ou  
5 analogue.

Il sera mieux compris à la lecture de la description ci-après et à l'observation des dessins dans lesquels :

- la figure 1 représente un schéma illustratif du  
10 procédé, objet de la présente invention, mis en œuvre entre une entité prouveur et une entité vérifieur ;

- la figure 2a représente un schéma illustratif du  
procédé, objet de la présente invention, mis en œuvre à  
partir d'un algorithme de Rabin en vérification  
15 d'authentification ;

- la figure 2b représente un schéma illustratif du  
procédé, objet de la présente invention, mis en œuvre à  
partir d'un algorithme de Rabin en vérification de  
signature ;

20 - la figure 3a représente un schéma illustratif du  
procédé, objet de la présente invention, mis en œuvre à  
partir d'un algorithme RSA en vérification  
d'authentification ;

- la figure 3b représente un schéma illustratif du  
25 procédé, objet de la présente invention, mis en œuvre à  
partir d'un algorithme RSA en vérification de signature.

Une description plus détaillée du procédé, objet de l'invention, sera donnée en liaison avec la figure 1 et les figures suivantes.

30 Le procédé objet de l'invention met en œuvre, au  
niveau de l'entité vérifieur, des algorithmes à clé  
publique requérant des multiplications modulo  $n$ , ou des  
grands nombres, et les modifie légèrement en faisant faire

le calcul d'un ou de plusieurs quotients  $q$  à l'extérieur, c'est-à-dire au niveau de l'entité prouveur, et en fournissant ce ou ces quotients au vérifieur. Ainsi le vérifieur peut plus facilement et plus rapidement calculer certaines multiplications modulaires : au lieu de calculer  $a*b$  modulo  $n$ , il aura juste à calculer  $a*b$ ,  $q*n$ , et  $a*b-q*n$ ,  $a$ ,  $b$  désignant des valeurs des calcul de vérification de signature ou d'authentification. Parfois, pour la sécurité il utilise cette dernière valeur d'une façon qui lui permettra de s'assurer que cette dernière valeur est bien comprise entre 1 et  $n$ . Lorsque l'on modifie ainsi un algorithme, en "précalculant" donc certains quotients, qui sont fournis au vérifieur afin de simplifier les calculs exécutés par ce dernier, on parle d'algorithme "sous-jacent" pour désigner l'algorithme initial dont on est parti, avant de faire cette modification. Ainsi, en référence à la figure 1, conformément à un aspect remarquable du procédé objet de la présente invention, le ou les quotients  $q$ , vérifiant la relation  $q=a*b/n$ , constituent une ou plusieurs valeurs de prévalidation transmises à l'entité vérifieur afin de permettre à l'entité vérifieur d'effectuer au moins une réduction modulaire en l'absence de toute opération de division pour cette réduction modulaire. En référence à la figure 1, on indique que le procédé objet de l'invention peut être mis en œuvre soit en vérification de l'authentification, suite à l'envoi d'une valeur d'incitation tel qu'un aléa  $a$  (voir la référence 0 sur la figure), calcul (référence 1) en interne au niveau du prouveur d'une valeur de réponse  $b = a^d \text{ mod } n$ , et de la valeur de prévalidation  $q$ , transmission (référence 2) de  $b$  et  $q$  du prouveur au vérifieur et calcul (référence 3) par le vérifieur des quantités  $a*b$ ,  $q*n$  et  $a*b-q*n$  pour

procéder à la vérification de l'authentification, soit à la vérification de signature d'un message M, suite au calcul (référence 1) au niveau du prouveur d'une signature  $S = S_d(M)$  du message M et de la valeur de prévalidation q, envoi (référence 2) du vérifieur au prouveur de q, S et M, calcul (référence 3) au niveau du vérifieur des quantités  $a*b = S*S$ ,  $q*n$  et  $a*b-q*n$  pour procéder à la vérification de signature.

Dans la figure 1 et les figures suivantes, une flèche droite représente la transmission des valeurs précitées entre vérifieur et prouveur ou réciproquement et une boucle fléchée au niveau du prouveur ou du vérifieur représente la mise en œuvre d'un calcul interne au niveau du prouveur ou du vérifieur. Enfin, dans la suite de la description, on désigne par réponse R soit la valeur calculée b par chiffrement de l'aléa a dans le cas d'une vérification d'authentification  $b = a^d \text{ mod } n$ , soit la valeur de signature  $S = S_d(M)$  suite à la mise en présence du vérifieur et du prouveur.

Différents exemples de mise en œuvre du procédé objet de la présente invention seront maintenant décrits à partir des algorithmes sous-jacents, désignés par algorithmes RSA et algorithmes de Rabin.

#### Algorithmes RSA et de Rabin sous-jacents

L'algorithme RSA est le plus célèbre des algorithmes cryptographiques asymétriques. Il a été inventé par RIVEST, SHAMIR et ADLEMAN en 1978. On peut le trouver décrit dans :

R.L. RIVEST, A. SHAMIR, L.M. ADLEMAN : A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM, 21, n°2, 1978, pp. 120-126. ou dans les documents suivants :

- ISO/IEC 9594-8/ITU-T X.509, Information Technology - Open Systems Interconnection - The Directory: Authentication Framework ;
- ANSI X9.31-1, American National Standard, Public-Key Cryptography Using Reversible Algorithms for the Financial Services Industry, 1993.

Ces documents sont introduits dans la présente description à titre de référence.

L'algorithme RSA utilise un nombre entier  $n$  qui est le produit de deux grands nombres premiers  $p$  et  $r$ , et un nombre entier  $e$ , premier avec  $\text{ppcm}(p-1, r-1)$ , et tel que  $e \neq \pm 1$  modulo  $\text{ppcm}(p-1, r-1)$ . Les entiers  $n$  et  $e$  constituent la clé publique. Le calcul en clé publique fait appel à la fonction  $\alpha$  de  $\mathbb{Z}/n\mathbb{Z}$  dans  $\mathbb{Z}/n\mathbb{Z}$  définie par  $\alpha(x) = x^e \bmod n$ . Le calcul en clé secrète fait appel à la fonction  $\alpha^{-1}(y) = y^d \bmod n$ , où  $d$  est l'exposant secret, appelé aussi "clé secrète" ou "clé privée", défini par  $ed \equiv 1 \bmod \text{ppcm}(p-1, r-1)$ .

Notons  $n$  le modulo public RSA, notons  $d$  l'exposant secret RSA et notons  $e$  l'exposant public RSA.

Dans le cas d'une vérification d'authentification, le vérifieur génère un nombre aléatoire  $A$  modulo  $n$ , et l'envoie au prouveur. Celui-ci calcule alors  $B = A^d \bmod n$ , et renvoie cette valeur  $B$  au vérifieur. Celui-ci accepte alors l'authentification si et seulement si:  $B^e \bmod n = A$ .

La plus petite valeur de  $e$  pour mettre en œuvre l'algorithme RSA est  $e = 3$ . Pour  $e = 2$ , on parle d'algorithme de Rabin ; celui-ci sera décrit ci-après dans la description. Cette valeur  $e = 3$  est intéressante car elle permet au vérifieur de n'avoir à effectuer que deux multiplications modulaires.

L'algorithme de Rabin est en quelque sorte un algorithme RSA avec l'exposant public  $e = 2$ . En fait, lorsque  $e = 2$ , la fonction  $x^e$  n'est pas bijective modulo  $n$ , lorsque  $n$  est le produit de deux nombres premiers  $> 2$ , on introduit donc des petites modifications dans l'utilisation de l'algorithme de Rabin par rapport au RSA.

On peut trouver une description de l'algorithme de Rabin dans :

10 M.O. Rabin, Digitized Signatures and Public-Key Functions as intractable as Factorization, Technical Report LCS/TR-212, M.I.T. Laboratory for Computer Science, 1979, introduit dans la présente demande de brevet à titre de référence.

15 Exemples de mise en œuvre du procédé objet de l'invention à partir des algorithmes de Rabin et RSA

♦ Algorithme de Rabin

Le procédé, objet de la présente invention, sera tout d'abord décrit dans un mode de réalisation particulier non limitatif à partir de l'algorithme de Rabin, soit pour  $e = 2$ .

♦♦ Vérification d'authentification

Ainsi que représenté en figure 2a, un exemple possible d'utilisation de l'algorithme de Rabin en vérification d'authentification est maintenant décrit.

Notons  $n$  le modulo public. Le vérifieur génère un nombre aléatoire  $A$  modulo  $n$ , et l'envoie, (référence 0 sur la figure), au prouveur. Celui-ci calcule alors un nombre  $B$  (référence 1), et renvoie cette valeur  $B$  au vérifieur.

30 Celui-ci accepte alors l'authentification si et seulement si:  $B*B$  modulo  $n$  est égal à l'une des quatre valeurs possibles suivantes :  $A$ , ou  $n-A$ , ou  $C*A$  modulo  $n$ , ou  $-C*A$

modulo  $n$ .  $C$  est un nombre fixé par le protocole,  $C = 2$  le plus souvent.

Pour simplifier le processus de vérification, conformément au procédé objet de la présente invention, le  
 5 prouveur n'envoie pas, (référence 2), la valeur  $B$  seule : il envoie  $B$  et  $Q$ , où  $Q$  est le quotient de  $B*B$  par le modulo public  $n$ . Le vérifieur vérifie alors que  $D_{AR} = B*B - Q*n$  est bien égal à l'une des quatre valeurs suivantes :  $A$ ,  $n-A$ ,  $(C*A)$  modulo  $n$ , ou  $(-C*A)$  modulo  $n$ . De plus, il  
 10 peut calculer  $(C*A)$  modulo  $n$  en calculant  $C*A$ , en gardant cette valeur si elle est  $< n$ , et en prenant la valeur  $C*A - n$  sinon. De même, il peut calculer  $(-C*A)$  modulo  $n$  en calculant  $n-C*A$ , en gardant cette valeur si elle est  $\geq 0$ , et en prenant la valeur  $C*n - C*A$  sinon. Ainsi le  
 15 vérifieur n'a plus aucune division à effectuer.

#### ♦♦ Vérification de signature

Ainsi que représenté en figure 2b, et en conservant les mêmes notations que ci-dessus, on note  $M$  le message dont le vérifieur souhaite vérifier la signature  
 20  $S$ . La signature  $S$  est obtenue à partir de la clé privée  $d$  par  $S = S_d(M)$ ,  $S_d(M)$  désignant l'opération de calcul de signature du message  $M$ . Si  $S$  est une signature Rabin de  $M$ , alors le vérifieur vérifie normalement que  $S*S$  modulo  $n = f(M)$  ou  $n-f(M)$ , ou  $(2*f(M)$  modulo  $n)$  ou  $(-2*f(M)$  modulo  
 25  $n)$ , où  $f$  est une fonction publique standardisée du message  $M$ . Par exemple  $f$  est la fonction identité, ou bien est décrite dans une norme de signature ; par exemple on peut utiliser les opérations de *paddage* ou concaténation de la norme PKCS#1, établie pour du RSA normalement, confer les  
 30 éléments descriptifs de cette norme ci-après dans la description.

En conservant les mêmes notations que ci-dessus, pour simplifier le processus de vérification de la

signature, ainsi que représenté en figure 2b, dans le procédé objet de la présente invention, le prouveur n'envoie pas, (référence 2), la valeur S seule : il envoie S et Q, où Q est le quotient de  $S*S$  par le modulo public n. Le vérifieur vérifie alors que  $D_{SR} = S*S - Q*n$  est bien égal à  $f(M)$ , ou  $n-f(M)$ , ou  $C*f(M)$  modulo n, ou  $-C*f(M)$  modulo n, où C est un nombre fixé par le protocole, C pouvant être pris égal à 2. Comme ces deux dernières valeurs peuvent être calculées modulo n en effectuant zéro ou une soustraction par n, le vérifieur n'a plus aucune division à calculer.

#### ◆ Algorithme RSA

Le procédé, objet de la présente invention, sera maintenant décrit dans un mode de réalisation particulier non limitatif à partir de l'algorithme RSA, soit pour  $e = 3$ .

#### ◆◆ Vérification d'authentification

Ainsi que représenté en figure 3a, à partir d'un aléa A, pour simplifier le processus de vérification, dans la présente invention le prouveur n'envoie pas, (référence 2), la valeur B seule : il envoie B, Q1 et Q2, où Q1 est le quotient de  $B*B$  par le modulo public n, et où Q2 est le quotient de  $B*(B*B - Q1*n)$  par n. Le vérifieur vérifiera alors que  $D_{ARSA} = B*(B*B - Q1*n) - Q2*n$  est bien égal à A. Ainsi le vérifieur n'a plus aucune division à effectuer.

#### ◆◆ Vérification de signature

En conservant les mêmes notations que ci-dessus et en notant M le message dont le vérifieur souhaite vérifier la signature S, S est une signature RSA de M, alors le vérifieur vérifie normalement que  $S^e$  modulo n =  $f(M)$ , où f est une fonction publique standardisée du message M. Par exemple f est la fonction identité, ou bien est décrite dans une norme de signature RSA, comme par exemple la

norme PKCS#1. La fonction publique normalisée peut  
consister à appliquer au message M une fonction de  
condensation SHA-1 pour obtenir un condensé de message CM,  
puis à concaténer à ce condensé de message une valeur  
5 constante.

Ainsi que représenté en figure 3b, et en  
conservant les mêmes notations que ci-dessus, pour  
simplifier le processus de vérification de la signature,  
dans le procédé, objet de la présente invention, le  
10 prouveur n'envoie pas, (référence 2), la valeur S seule :  
il envoie S, Q1 et Q2, où Q1 est le quotient de  $S^2$  par le  
modulo public n, et où Q2 est le quotient de  $S^2 - Q1 \cdot n$   
par n. Le vérifieur vérifiera alors que  $D_{\text{SRSA}} =$   
 $S \cdot (S^2 - Q1 \cdot n) - Q2 \cdot n$  est bien égal à  $f(M)$ . Ainsi le  
15 vérifieur n'a plus aucune division à effectuer.

La fonction de condensation SHA-1 est une fonction  
publique de "condensation". Elle prend en entrée un  
message dont la taille peut aller de 0 octets à plusieurs  
Giga octets, et donne en sortie un "condensé" du message  
20 de 160 bits. Cette fonction est souvent utilisée dans des  
normes ou avec des algorithmes de signature, car elle est  
réputée être résistante aux collisions, c'est-à-dire que  
l'on ne sait pas trouver concrètement deux messages  
distincts qui ont le même condensé (il en existe mais on  
25 ne sait pas comment trouver un tel couple de messages).  
Ceci permet de signer le condensé des messages plutôt que  
les messages eux-mêmes.

La norme PKCS#1 est une norme de signature RSA.  
Elle décrit une fonction publique f. Cette fonction f est  
30 appliquée sur le message M à signer avec RSA avant de  
lancer l'opération d'exponentiation modulaire RSA  
proprement dite : la signature RSA de M sera donc

$$S = (f(M))^d \text{ modulo } n, \text{ où } n \text{ est le modulo public}$$

RSA et où  $d$  est l'exposant secret RSA.  $f$  utilise une fonction de condensation (par exemple SHA-1) suivie d'un *paddage*, ou concaténation, avec une constante.

Pour une description plus détaillée, on peut  
5 consulter :

PKCS#1, *RSA Encryption Standard*, version 2, 1998,  
disponible à l'adresse suivante :

<ftp://ftp.rsa.com/pub/pkcs/doc/pkcs-1v2.doc>

dont la version éditée est introduite dans la présente  
10 demande à titre de référence.

L'invention consiste ainsi à fournir des données  
supplémentaires au vérifieur afin de lui faciliter les  
calculs. Pour précalculer ces données, ici des quotients  
constituant la ou les valeurs de pré-validation, on n'a  
15 pas besoin d'utiliser la clé secrète de l'algorithme. Cela  
signifie que ces données sont complètement redondantes par  
rapport aux valeurs transmises à la carte dans une  
utilisation "*classique*" de l'algorithme asymétrique. En  
fait, dans la version "*classique*", la carte sait retrouver  
20 elle-même ces quotients. Il n'y a donc aucune information  
supplémentaire fournie à la carte, au sens de la théorie  
de l'information, lorsqu'on met en œuvre le procédé, objet  
de la présente invention tel que décrit précédemment. Cela  
montre que la sécurité de l'ensemble n'est en rien  
25 affaiblie par rapport à la mise en œuvre "*classique*" de  
l'algorithme.

REVENDICATIONS

1. Procédé de vérification de signature, respectivement d'authentification, au moyen d'un processus de calcul cryptographique asymétrique à clé privée et à  
5 clé publique, entre une entité "prouveur" et une entité "vérifieur", l'entité prouveur effectuant des calculs cryptographiques à partir de ladite clé privée en vue d'effectuer un calcul de signature, respectivement d'une valeur d'authentification, constituant une valeur de  
10 réponse et l'entité vérifieur, à partir de cette valeur de réponse, effectuant des calculs cryptographiques à partir de ladite clé publique en vue de procéder à cette vérification de signature, respectivement cette authentification, les opérations de calcul cryptographique  
15 mettant en œuvre le calcul de multiplications modulo  $n$  ou des grands nombres, caractérisé en ce que pour un processus de calcul cryptographique mettant en œuvre une clé publique, comprenant un exposant public  $e$  et un modulo public  $n$ , et une clé privée comprenant un exposant privé, celui-ci comprend les étapes suivantes :

- calculer au niveau de ladite entité prouveur au moins une valeur de pré-validation ;
- transmettre de l'entité prouveur à l'entité vérifieur ladite au moins une valeur de pré-validation, cette valeur de pré-validation permettant à l'entité  
25 vérifieur d'effectuer au moins une réduction modulaire en l'absence de toute opération de division pour cette réduction modulaire.

2. Procédé selon la revendication 1, caractérisé  
30 en ce que pour un exposant public  $e=2$ , le processus de calcul cryptographique étant basé sur un algorithme de RABIN, ladite au moins une valeur de pré-validation comprend une valeur unique, quotient  $Q$  du carré de ladite

valeur de signature, respectivement de réponse, par ledit modulo public  $n$ ,  $Q = R^2/n$ , où  $R$  désigne ladite valeur de signature, respectivement de réponse, à une authentification.

5           3. Procédé selon la revendication 2, caractérisé en ce que suite à la réception par ladite entité vérifieur de ladite valeur de réponse à une vérification d'authentification respectivement de signature d'un message ( $M$ ) et de ladite au moins une valeur de pré-validation, comprenant ledit quotient, ce procédé  
10 comprend, au niveau de ladite entité vérifieur, les étapes suivantes :

- calculer la différence ( $D_{AR}$ ,  $D_{SR}$ ) entre le carré de la valeur de réponse  $R^2$  et le produit  $Q*n$  dudit  
15 quotient  $Q$  par ledit modulo public  $n$ ,

$$(D_{AR}, D_{SR}) = R^2 - Q*n ;$$

- vérifier l'égalité de ladite différence avec la valeur d'une fonction de cette valeur de réponse, en l'absence de toute opération de division par l'opération  
20 modulo  $n$ .

4. Procédé selon la revendication 1, caractérisé en ce que pour un exposant public  $e = 3$ , le processus de calcul cryptographique étant basé sur un algorithme RSA, ladite au moins une valeur de pré-validation comprend :

25           - un premier quotient  $Q_1$  du carré  $R^2$  de ladite valeur de réponse  $R$  par ledit modulo public  $n$  ;

- un deuxième quotient  $Q_2$  du produit de ladite valeur de réponse et de la différence entre le carré  $R^2$  de cette valeur de réponse et du produit dudit premier  
30 quotient  $Q_1$  et du modulo public  $n$ , par ledit modulo public  $n$ ,  $Q_2 = R*(R^2 - Q_1*n)/n$ .

5. Procédé selon la revendication 4, caractérisé en ce que suite à la réception de ladite valeur de réponse

R et de ladite au moins une valeur de pré-validation comprenant lesdits premier et deuxième quotients  $Q_1$ ,  $Q_2$ , ledit procédé comprend, au niveau de ladite entité vérifieur, les étapes suivantes :

- 5                   - calculer la différence ( $D_{ARSA}$ ,  $D_{SRSA}$ ) entre le produit de ladite valeur de réponse R et de la différence entre le carré  $R \cdot R$  de cette valeur de réponse et le produit dudit premier quotient  $Q_1$  et du modulo public n et le produit dudit deuxième quotient  $Q_2$  et dudit modulo public n, ( $D_{ARSA}$ ,  $D_{SRSA}$ ) =  $R \cdot (R \cdot R - Q_1 \cdot n) - Q_2 \cdot n$  ;

                  - vérifier l'égalité de cette différence avec la valeur d'une fonction de ladite valeur de réponse, en l'absence de toute opération de division par opération modulo n.

- 15               6. Procédé selon la revendication 3 ou 5, caractérisé en ce que pour une opération de vérification de signature d'un message (M), ladite fonction comprenant une fonction publique normalisée  $f(M)$  de ce message M, il comprend les étapes suivantes :

- 20                   - appliquer à ce message une fonction de condensation pour obtenir un condensé de message CM ;  
                  - concaténer à ce condensé de message une valeur constante.

- 25               7. Procédé selon l'une des revendications 3 ou 5, caractérisé en ce que, pour une opération de vérification d'authentification, ce procédé comprend en outre l'étape de transmission de l'entité vérifieur à l'entité prouveur d'une valeur d'incitation.

- 30               8. Procédé selon la revendication 7, caractérisé en ce que ladite valeur d'incitation comprend une valeur aléatoire A modulo n, ladite valeur de réponse R comprend une valeur chiffrée B, et ladite fonction de la valeur de

réponse comprend une fonction  $f(A)$  de ladite valeur aléatoire  $A$ .

9. Procédé selon l'une des revendications 3 et 7, caractérisé en ce que ladite fonction  $f(A)$  de ladite valeur aléatoire  $A$  comprend une fonction parmi les fonctions  $f(A) = A$ ,  $\bar{f}(A) = n-A$ ,  $f(A) = C \cdot A$  modulo  $n$ ,  $f(A) = -C \cdot A$  modulo  $n$ .

10. Procédé selon la revendication 9, caractérisé en ce que, au niveau de l'entité vérifieur, le calcul de ladite fonction  $f(A) = C \cdot A$  modulo  $n$  comprend le calcul de la valeur  $C \cdot A$  et la mémorisation de cette valeur si  $C \cdot A < n$ , et le calcul et la mémorisation de la valeur  $C \cdot A - n$  sinon, et en ce que le calcul de ladite fonction  $f(A) = -C \cdot A$  modulo  $n$  comprend le calcul de la valeur  $n - C \cdot A$  et la mémorisation de cette valeur si  $n - C \cdot A \geq 0$ , et sinon le calcul de la valeur intermédiaire  $C \cdot n - C \cdot A$ , et, si cette valeur intermédiaire est supérieure ou égale à zéro, le calcul et la mémorisation de la valeur de  $C \cdot n - C \cdot A$  comme valeur affectée à la valeur de  $-C \cdot A$  modulo  $n$ , ce qui permet de vérifier l'égalité de ladite authentification en l'absence de toute division pour la réduction modulaire.

11. Procédé selon les revendications 5 et 8, caractérisé en ce que ladite fonction  $f(A)$  de ladite valeur aléatoire  $A$  est la fonction  $f(A) = A$ , ce qui permet de vérifier l'égalité de ladite différence et la validité de ladite authentification, en l'absence d'opération de division pour la réduction modulaire.

12. Procédé selon la revendication 1, caractérisé en ce que ladite valeur de réponse, valeur chiffrée  $B$ , et ladite valeur de quotient  $Q$  sont concaténées préalablement à leur transmission de l'entité prouveur à l'entité vérifieur.

13. Utilisation du procédé selon la revendication 1, l'entité vérifieur comprenant un système embarqué tel qu'une carte à microprocesseur et l'entité prouveur un système lecteur de système embarqué.

### ABREGE DESCRIPTIF

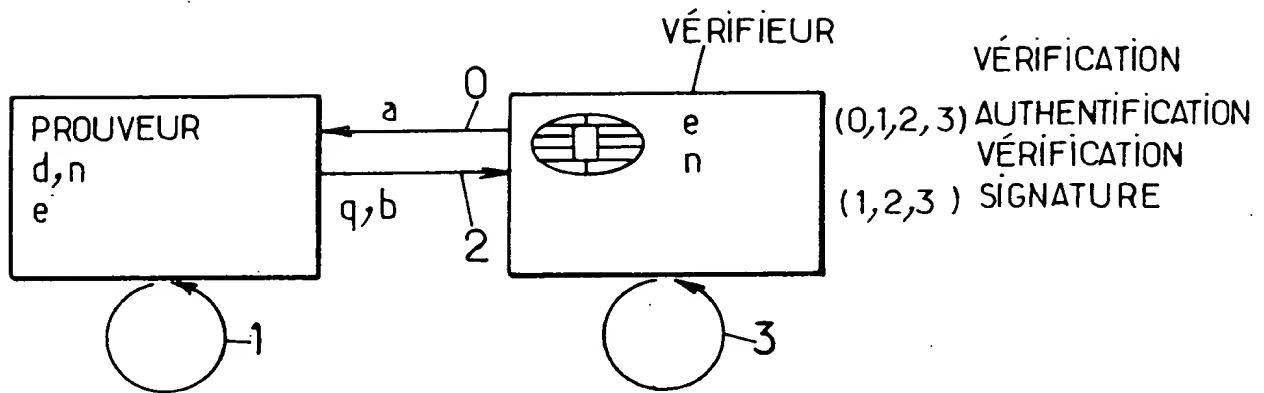
L'invention concerne un procédé de vérification de signature ou d'authentification entre prouveur et vérifieur à partir d'un algorithme de calcul cryptographique asymétrique.

Le prouveur calcule (1) au moins une valeur de prévalidation  $q$ , quotient de deux valeurs cryptographiques  $a$ ,  $b$  par le modulo public  $n$ , et transmet au vérifieur cette valeur  $q$ . Le vérifieur calcule (3) les produits  $a*b$  et  $q*n$  et la différence  $a*b-q*n$  pour effectuer au moins une réduction modulaire en l'absence d'opération de division.

Application à la vérification de signature ou d'authentification entre un prouveur, micro-ordinateur, et un vérifieur, carte à microprocesseur.

Figure 1.

1/3



$$q = a * b / n$$

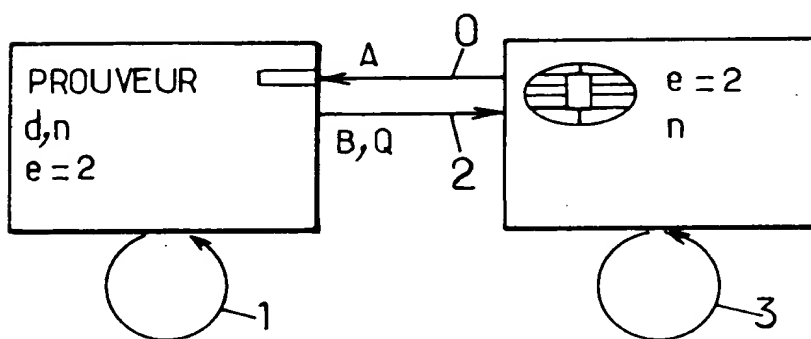
$$b = \begin{cases} a^d \bmod n & \text{si } (0,1,2,3) \\ S = S_d(M) & \text{si } (1,2,3) \end{cases}$$

$$a * b$$

$$q * n$$

$$a * b - q * n$$

FIG.1.



$$R = B = A^d \bmod n$$

$$Q = B * B / n$$

$$D_{AR} = B * B - Q * n$$

$$D_{AR} = A$$

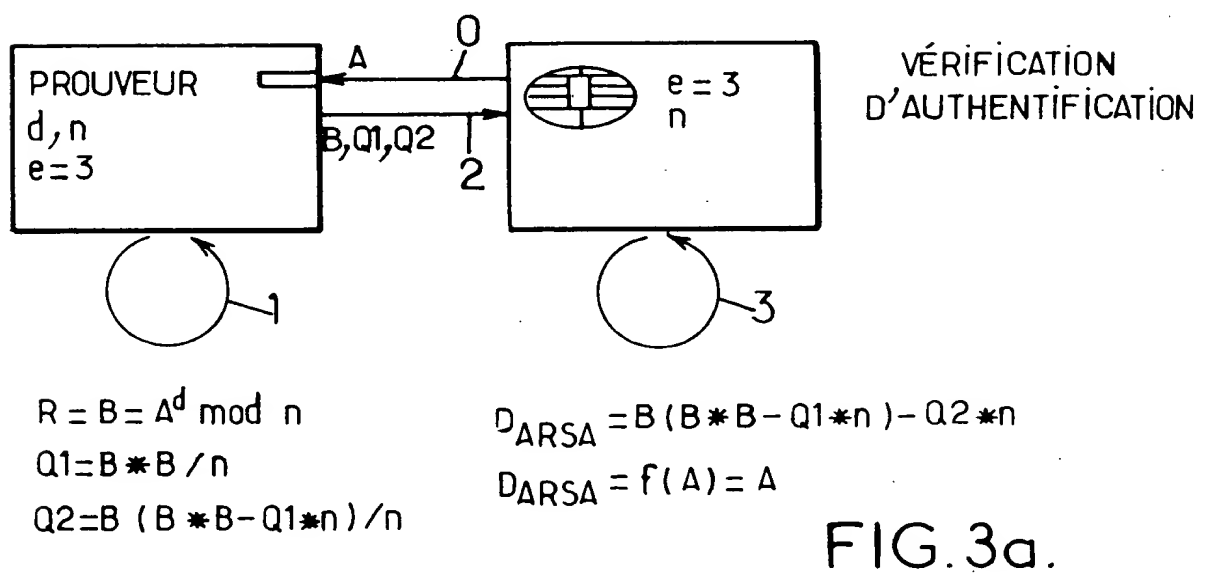
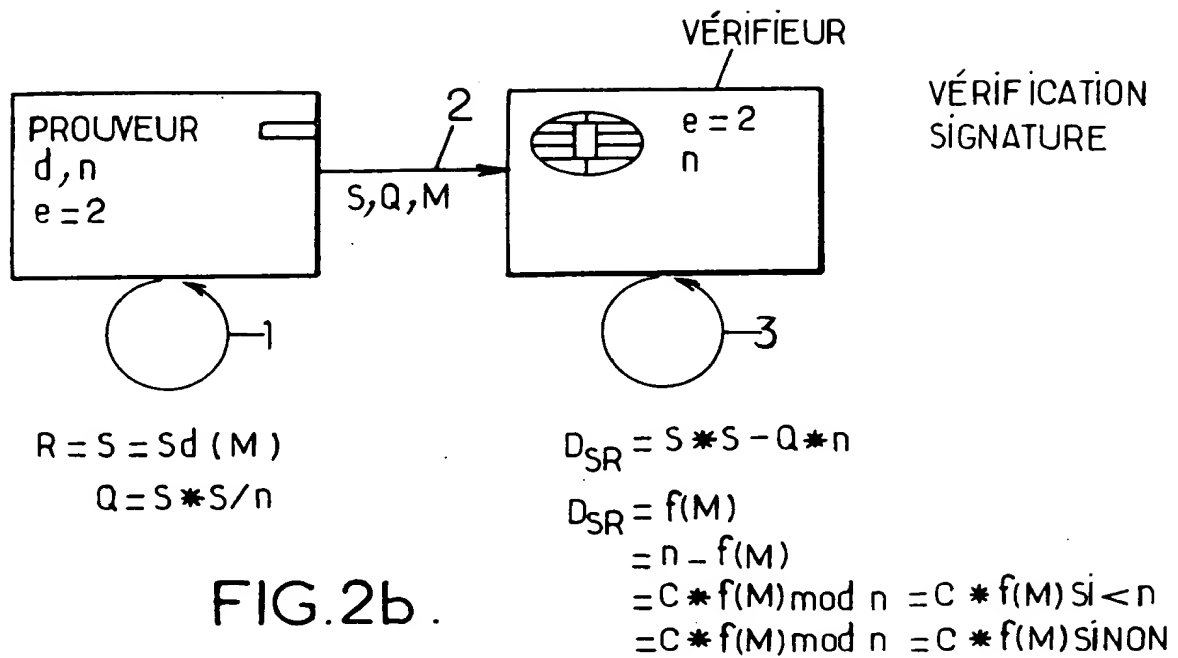
$$D_{AR} = n - A$$

$$\left. \begin{aligned} D_{AR} &= C * A \bmod n \\ D_{AR} &= -C * A \bmod n \end{aligned} \right\} = C * A \text{ si } C * A < n$$

$$= C * A - n \text{ SINON}$$

FIG. 2a.

2/3



3/3

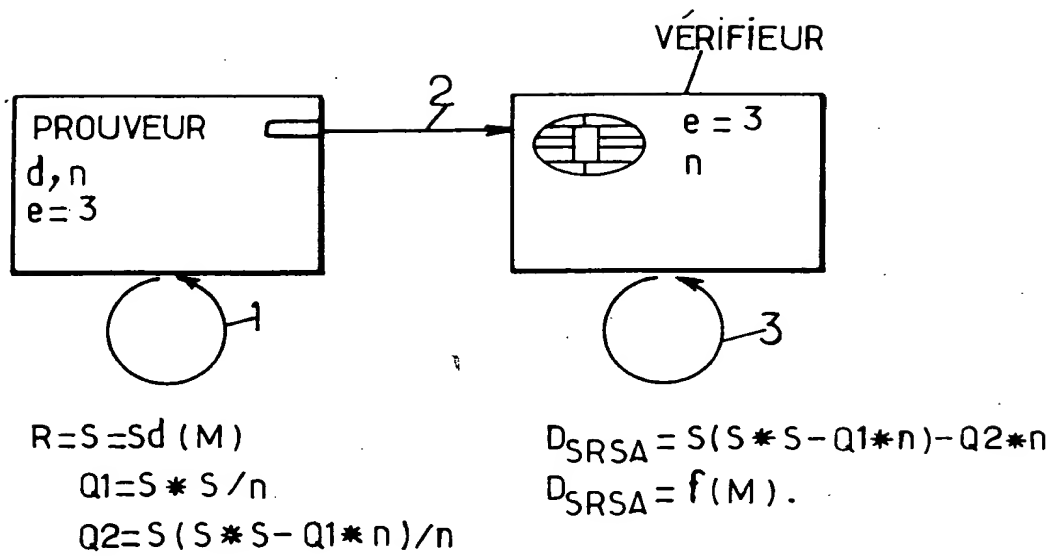


FIG.3b.